

УДК 621.396.62+658.56

Formation of the Integrated Information Environment of a Radio-Electronic Equipment Design

Oleg V. Drozd* and Denis V. Kapulin

Siberian Federal University
79 Svobodny, Krasnoyarsk, 660041, Russia

Received 12.04.2017, received in revised form 24.04.2018, accepted 27.02.2019

Widely known method of combining of different information instruments of design support is formation of a common information space. However the modern nomenclature of software for the organization of a common information space not always considers specifics of a radio-electronic equipment production. In article the task on development of a mathematical model for the electronic designer document and formalization of interaction between design tools at all stages of a project cycle of a radio-electronic equipment is researched. The mathematical model of the electronic designer document with using of the subject-oriented ontology apparatus and theory of sets is offered. Dynamics of the electronic designer document life cycle is described by using of the automata theory, graphically the model of dynamics is presented in the form of the oriented graph, simulation modeling of dynamics was carried out in the numerical computing environment MATLAB with using of an extension packet Stateflow. Application area of use of the developed mathematical model is design and tests of digital receivers of navigation satellite signals. The hardware-software automated bench and software of interaction with management system the engineering data "Lotsman: PLM" within a common information space are developed for implementation of tests of the designed digital receiver.

Keywords: PDM, design support, common information space, subject-oriented ontology, theory of sets, automata theory, testing of a radio-electronic equipment.

Citation: Drozd O.V., Kapulin D.V. Formation of the integrated information environment of a radio-electronic equipment design, J. Sib. Fed. Univ. Eng. technol., 2019, 12(3), 293-313. DOI: 10.17516/1999-494X-0137.

© Siberian Federal University. All rights reserved

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

* Corresponding author E-mail address: olvldrozd@gmail.com

Формирование интегрированной информационной среды поддержки проектирования радиоэлектронной аппаратуры

О.В. Дрозд, Д.В. Капулин

*Сибирский федеральный университет
Россия, 660041, Красноярск, пр. Свободный, 79*

Широко известным способом объединения различных информационных средств поддержки проектирования является формирование единого информационного пространства. Однако современная номенклатура программных средств организации единого информационного пространства не всегда учитывает специфику производства радиоэлектронной аппаратуры. В статье исследуется задача по разработке математической модели электронного конструкторского документа и формализации взаимодействия между средствами проектирования на всех этапах проектного цикла радиоэлектронной аппаратуры. Предлагается математическая модель конструкторского документа с использованием аппарата предметно-ориентированных онтологий и теории множеств. Динамика жизненного цикла конструкторского документа описывается с помощью теории автоматов, графически модель динамики представлена в виде ориентированного графа, имитационное моделирование динамики проводилось в среде MATLAB с использованием пакета Stateflow. Прикладной областью применения разработанной математической модели является проектирование и испытания цифровых приемников навигационных спутниковых сигналов. Для осуществления испытаний цифрового приемника разработаны программно-аппаратный автоматизированный стенд и программные средства взаимодействия стенда с системой управления инженерными данными «Лоцман: PLM» в рамках единого информационного пространства.

Ключевые слова: поддержка проектирования, единое информационное пространство, предметно-ориентированная онтология, теория множеств, теория автоматов, испытание радиоэлектронной аппаратуры.

Введение

Разработка современной радиоэлектронной аппаратуры связана с непрерывным увеличением степени сложности проектируемых устройств, ужесточением требований к технологическому процессу полупроводникового производства, эволюцией методов и подходов к проектированию. При проектировании сложных цифровых систем широко применяются методологии «Система на кристалле» (*System-on-a-Chip, SoC*) и «Сеть на кристалле» (*Network-on-a-Chip, NoC*), ориентированные на использование заранее разработанных схмотехнических блоков (программных процессоров, блоков памяти, цифровых усилителей, фильтров и т. д.) [1–3]. На каждом этапе такого процесса используется собственная система автоматизированного проектирования, что повышает эффективность отдельных этапов или стадий процесса проектирования. Но при этом неизбежно образуется поливендорная среда, содержащая разнородные решения нескольких производителей (вендоров), что порождает ряд проблем, связанных с отсутствием единых информационных решений:

- отсутствие цельного системного представления о процессе проектирования;

- невозможность детального анализа всей совокупности проектной информации;
- сложность быстрого принятия проектных решений из-за отсутствия унифицированных подходов к проектированию и обработке проектной информации.

Кроме того, стоит учитывать особенности современного производства радиоэлектронной аппаратуры: большой ассортимент выпускаемой продукции, мелкосерийный характер производства, необходимость быстрой переналадки и перенастройки как всего производства, так и процессов проектирования, обусловленная выпуском новой продукции и/или заменой компонентной базы. В таких условиях поливендорная среда проектирования, характеризующаяся наличием значительного числа как систем автоматизированного проектирования, так и иных прикладных программ различных производителей, нуждается в пересмотре с точки зрения эффективности организации применения разнородных информационно-программных решений.

Известным способом разрешения недостатков поливендорной среды является объединение различных информационных, программных и аппаратных средств поддержки процессов проектирования в единый комплекс информационно-программных решений и формирование на его основе единого информационного пространства [4, 5]. Информационное взаимодействие по такому принципу должно быть поддержано автоматизированными информационными системами сбора, обработки и анализа проектной информации, представленной в виде различных электронных конструкторских документов. Данный подход уже на ранних этапах проектирования обеспечивает формализацию методов и протоколов взаимодействия между аппаратными и программными средствами различных производителей, что существенно повышает эффективность обработки информации и способствует быстрому принятию проектных решений.

Постановка задачи

Учитывая современные подходы к проектированию, разработке и организации производства радиоэлектронной аппаратуры, использовать ограниченную, нерасширяемую номенклатуру прикладных программных продуктов для решения разнообразных проектных задач невозможно. При этом наличие большого количества разнородных, зачастую несовместимых между собой программных систем порождает значительное число плохо управляемых потоков информации, нуждающихся в унификации и единых подходах к их анализу и обработке. Единое информационное пространство (ЕИП), формируемое для систематизации и унификации обработки информации, связанной с решением прикладных задач проектирования и производства радиоэлектронной аппаратуры, позволит исключить недостатки поливендорной среды проектирования.

Формирование ЕИП или интегрированной информационной среды невозможно без разработки математической модели электронного конструкторского документа (ЭКД) и формализации взаимодействия между программно-аппаратными средствами проектирования на всех этапах проектного цикла (ПЦ). Формальное описание ЭКД должно включать в себя модели структуры и динамики процессов жизненного цикла (ЖЦ) ЭКД, описывающие, соответственно, структурную организацию ЭКД, и переходы по стадиям жизненного цикла документов (состояния ЭКД). Таким образом, при формировании ЕИП следует:

- формализовать описание ЭКД в виде единой математической модели, сочетающей в себе представление электронного конструкторского документа в рамках ЕИП с точки зрения его структуры и поведения;

- разработать требования к единому формату ЭКД, обеспечивающему эффективный обмен данными между средствами проектирования в ходе ПЦ;
- предложить методы и средства для интеграции программно-аппаратных комплексов проектирования в единую информационную среду с использованием единого формата ЭКД.

Стоит отметить, что уже известные решения в области организации ЕИП успешно применяются при создании безлюдных гибких производственных систем в различных областях машиностроения, в том числе с использованием виртуальных автоматизированных систем управления производством, но единых, хорошо апробированных подходов, методов и средств, полностью удовлетворяющих потребности проектных организаций при разработке и мелкосерийном производстве сложной радиоэлектронной аппаратуры, пока нет [6–9].

Формализация структуры электронного конструкторского документа в рамках единого информационного пространства

Прежде чем перейти непосредственно к рассмотрению единой модели представления данных об изделии и структуры *PDM*-системы, необходимо кратко обрисовать маршрут проектирования СБИС класса «Система на кристалле». Весь цикл проектирования можно представить в виде совокупности пяти основных этапов [10, 11]:

1. Системное проектирование – определение основных эксплуатационно-технических свойств будущей системы. На основании этих свойств создается системная спецификация, которая может выступать частью технического задания на разработку системы.
2. Функциональное проектирование – разработка и верификация синтезируемой *RTL*-модели (*Register Transfer Level*) на уровне регистровых передач.
3. Логическое проектирование – разработка логической схемы СнК.
4. Топологическое проектирование – разработка и верификация топологии СБИС.
5. Этап верификации и подготовки к производству.

На этапе системного проектирования разрабатывается поведенческая модель СнК и определяется состав СФ-блоков. Исполняемые спецификации представляются в определенном формате на языках C, C++, *Verilog* и *VHDL*.

Разработка СнК начинается с анализа задач и требований, а также написания системной спецификации. При этом определяются основные эксплуатационно-технические свойства СнК, такие как требуемое быстродействие и допустимая потребляемая мощность.

Алгоритм работы СнК на уровне математического описания создается на основе разработанной системы спецификации. Производится математическое моделирование разработанных алгоритмов функционирования СнК, оценка требуемого быстродействия, разрядности и оценка формата представления внутренних данных. Также при необходимости может производиться синтез наборов данных (сигналов), предназначенных для тестирования схемотехнических решений СнК.

На основе алгоритма работы СнК разрабатывается поведенческая модель системы на уровне СФ-блоков в виде блок-схемы, отражающей принцип взаимодействия СФ-блоков в составе СнК и включающей их основные параметры. Для верификации поведенческой модели создают

проект тестового окружения (*Testbench*). Он содержит в себе тестовые последовательности, генераторы входных сигналов и средства отображения выходной информации. На основе тестового окружения разрабатываются последовательности для верификации проекта на нижних уровнях проектирования, а также для функционального тестирования опытных образцов. После этого проводят верификацию поведенческой модели путем компьютерного моделирования с помощью специальных программных средств.

Далее принимается решение о том, какие СФ-блоки поведенческой модели будут реализованы на аппаратном уровне, а какие – на программном, на основе встроенных в СнК процессорных ядер; разрабатывается интерфейс между аппаратным и программным обеспечениями и определяется общая аппаратная архитектура СнК. Разрабатывается спецификация и определяются приобретаемые, имеющиеся в наличии и вновь разрабатываемые СФ-блоки. В итоге создается набор спецификаций на разработку как программного обеспечения, так и каждого аппаратно реализуемого СФ-блока.

При совместной программно-аппаратной верификации проверяется функционирование аппаратного обеспечения, разрабатываемого в соответствии с имеющимися спецификациями, под управлением встроенного программного обеспечения в реальном масштабе времени. В качестве аппаратного обеспечения используются исполняемые спецификации аппаратно реализуемых блоков, в качестве программного – прототип прикладного программного обеспечения (ПО), развертываемого на базе СнК (например, прикладное ПО для цифровой обработки сигналов на базе операционных систем реального времени *Unison* и других *Unix*-подобных систем для архитектур *MicroBlaze* и *Cortex*).

При выборе архитектуры одним из ключевых параметров является низкое энергопотребление микросхемы. Исходя из этого, производится подбор СФ-блоков и разделение на аппаратный и программный уровни. Поэтому необходимо получить и систематизировать полную техническую информацию о покупных и имеющихся в наличии СФ-блоках, так как их характеристики могут существенно повлиять на архитектуру микросхемы.

Концептуальная модель единого информационного пространства

Рассмотрим концептуальную модель ЕИП, приведенную на рис. 1. Модель ЕИП (поз. 1) включает следующие основные составляющие:

- система управления данными об изделии (*PDM*-система, поз. 2) – информационная система, реализующая совокупность математических моделей ПЦ, обеспечивающая единый обменный формат ЭКД и предоставляющая платформу для информационного взаимодействия между поставщиками и потребителями информации;
- участники проектного цикла (поз. 3) – поставщики и потребители информации;
- аппаратные источники ЭКД (поз. 4) – стационарные и портативные рабочие станции, измерительное оборудование, отладочные средства различного типа;
- программные источники ЭКД (поз. 5) – развернутые на базе аппаратных источников системы автоматизированного проектирования (САПР), средства имитационного моделирования.



Рис. 1. Концептуальная модель ЕИП

Fig. 1. The Unified Information Space (UIS) conceptual model

Участники проектного цикла (поставщики информации) передают ЭКД, полученные на текущем этапе ПЦ, участникам-потребителям, которые могут быть расположены на текущем, последующем или предыдущем этапе ПЦ. Причем поставщик информации может также выступать и в роли потребителя информации от других участников ПЦ. Приведение разнородных ЭКД от программных и аппаратных систем сторонних разработчиков к единому формату осуществляется программным конвертером-преобразователем. Хранение ЭКД и информационное взаимодействие между участниками проектного цикла обеспечиваются *PDM*-системой.

Для формализации структуры ЭКД применим аппарат предметно-ориентированных онтологий и соответствующие разделы теории множеств [10–12]. Модель структуры ЭКД для такого случая определим следующим образом (1):

$$O = \langle C, I, E, L, F_{ext}, F_{md}, F_s, F_{exc} \rangle,$$

где C – набор категорий (классов объектов, понятий) ЕИП; I – набор информационных ресурсов (ЭКД), которые описываются различными категориями; E – единый обменный формат ЭКД, который позволяет формально описать все виды разнородных информационных ресурсов; L – набор внешних форматов представления ЭКД в ЕИП; $F_{ext}(L_i) \rightarrow E$ – функция преобразования ЭКД из внешних форматов в единый формат; $F_{md}(E) \rightarrow I$ – функция преобразования ЭКД из единого формата в набор информационных ресурсов; $F_s(I_i) \rightarrow I$ – функция поиска требуемых информационных ресурсов (I), удовлетворяющих заданным поисковым критериям (I_i); $F_{exc}(I_i) \rightarrow I_j$ – функция информационного обмена между узлами ЕИП.

Электронный конструкторский документ (I) как информационный ресурс можно представить в виде кортежа (2):

$$I = \langle p, c, v, d, S, F, f_t, f_m \rangle,$$

где $p = \{p_i\}$ – множество атрибутов документа, $i = 1 \div i$, i – количество типов атрибутов в информационном ресурсе; $c = \{c_j\}$ – множество категорий документов, $j = 1 \div j$, j – количество типов категорий, к которым может принадлежать информационный ресурс; $v = \{v_k\}$ – множество значений атрибутов, при этом $V = \{v_k \mid v_k \text{ – описание } k\text{-го значения атрибута, } k = 1 \div n, n \text{ – количество значений атрибутов в информационном ресурсе}\}$; $d = \{d_m\}$ – множество прав доступа к документу, при этом $D = \{d_m \mid d_m \text{ – описание } m\text{-го права доступа некоторого пользователя – чтения, записи, удаления, передачи; } m = 1 \div m, m \text{ – количество определенных прав доступа}\}$; S – набор настроек синхронизации, где каждая настройка определяет достаточные данные для осуществления информационного обмена, т. е. для выполнения функции $F_{exc}(I_i) \rightarrow I_j$, при этом $S = \{s_i \mid s_i \text{ – описание настройки синхронизации с } i\text{-м узлом ЕИП, } i = 1 \div k, k \text{ – количество узлов, с которыми синхронизируется информационный ресурс}\}$; F – набор прикрепленных файлов, при этом $F = \{f_i \mid f_i \text{ – описание } i\text{-го файла, } i = 1 \div l, l \text{ – количество прикрепленных к информационному ресурсу файлов}\}$; $f_t = \{f_h\}$ – множество типов файлов, $h = 1 \div h$, h – количество типов файлов, ассоциированных с информационным ресурсом; $f_m = \{m_p\}$ – множество атрибутов файлов, $p = 1 \div p$, p – количество значений атрибутов файлов, ассоциированных с информационным ресурсом.

Все создаваемые, модифицируемые и передаваемые в процессе проектирования радиоэлектронной аппаратуры электронные конструкторские документы можно разделить на пять основных категорий (v_k):

- v_1 – цифровая модель (модель в формате *MATLAB/Simulink*);
- v_2 – исходный код (описание устройства на языке *VHDL*);
- v_3 – отчет (отчет о верификации);
- v_4 – текстовый документ (текст спецификации);
- v_5 – электронный документ свободной формы (документы, не подпадающие под вышеперечисленные типы).

Документы категорий v_2 , v_3 и v_4 относятся к текстовым электронным документами, v_1 и v_5 – к нетекстовым. Каждой категории соответствует свой тип файлов (f_h): цифровая модель (f_1), исходный код (f_2), отчет (f_3), текстовый документ (f_4), электронный документ свободной формы (f_5).

В терминах предметно-ориентированной онтологии информационный ресурс состоит из реквизитной части (соответствует области метаданных ЭКД, определяется параметрами p, c, v, d, S) и содержательной части (параметры F, f_t, f_m). Атрибуты различных типов файлов (f_m) содержательной части в целом аналогичны (имя, размер и тип файла в терминах ПЦ и т. д.). Любой атрибут может заполняться пользователем в процессе работы с системой вручную путем выбора из списка доступных значений атрибута или автоматически при загрузке соответствующего файла в хранилище с преобразованием в единый формат.

Область метаданных представлена индивидуальной карточкой ЭКД. Набор атрибутов индивидуальной карточки документа зависит от того, куда передается или откуда получен ЭКД.

Таким же образом, как и при работе с файлами, любой атрибут может заполняться пользователем вручную или автоматически. Введение автоматического формирования метаданных для ЭКД позволяет снизить влияние человеческого фактора на документооборот в ходе проектного цикла [13–15].

Динамическая модель электронного конструкторского документа

Для полноты формализованного представления следует рассмотреть модель, описывающую особенности поведения ЭКД как некоторого динамического объекта. Для этого целесообразно представить модель переходов между стадиями жизненного цикла ЭКД в виде конечного автомата (КА) Мили (3):

$$S = \langle Q, D, V, W, N_0, X, Y, Z \rangle.$$

Описание конечного автомата включает в себя следующие элементы:

- $Q = \{q_\beta\}$, где $\beta = 1 \div \beta'$ – множество сигналов на изменение стадии ЖЦ ЭКД, поступающих в ЕИП на данном этапе ПЦ (входной алфавит);
- $D = \{d_\lambda\}$, где $\lambda = 1 \div \lambda'$ – множество ответных реакций на изменение стадии ЖЦ ЭКД на данном этапе ПЦ (выходной алфавит);
- $V = \{v_\gamma\}$, где $\gamma = 1 \div \gamma'$ – множество состояний ЭКД (алфавит состояний);
- $W = Q \times D \times V_1 \times V_2$ – множество действий, совершаемых над ЭКД, где четверка $(q_\beta, d_\lambda, v_{\gamma+1}, v_\gamma)$ означает, что документ по запросу q_β с ответом d_λ перешел из состояния v_γ в состояние $v_{\gamma+1}$;
- $N_0 = \{0, 1, 2 \dots\}$ – множество значений номеров этапов ЖЦ ЭКД;
- X – множество функций $x: N_0 \rightarrow Q$, задающих все возможные последовательности запросов к ЕИП в ходе ЖЦ ЭКД;
- Y – множество функций $y: N_0 \rightarrow D$, задающих все возможные последовательности ответов ЕИП по запросам (функция выходов);
- Z – множество функций $z: N_0 \rightarrow V$, задающих все возможные последовательности состояний к ЕИП в ходе ЖЦ ЭКД (функция переходов).

Для каждого документа, формируемого в ходе ПЦ, можно выделить последовательность стадий его собственного ЖЦ, что отражается в свойствах документа как «Стадия ЖЦ документа» (алфавит состояний КА, табл. 1).

Рассмотрим логическую последовательность выполнения процессов стадий жизненного цикла ЭКД в ЕИП (переходов конечного автомата S):

1. До начала проектирования документ находится в стадии ЖЦ «Новый».
2. При инициализации процесса проектирования документ переходит в стадию ЖЦ «В разработке».
3. Для предоставления общего доступа к документу его следует опубликовать в *PDM*-системе, при этом стадия жизненного цикла документа меняется на «Опубликован».
4. В случае изменения документ переходит в стадию ЖЦ «Внесение изменений в документ» и становится недоступным на время выполнения работ стадии.

Таблица 1. Стадии жизненного цикла ЭКД

Table. 1. Life cycle stages of the Electronic Design Document (EDD)

Обозначение	Стадия жизненного цикла
c_1	Новый
c_2	В разработке
c_3	Опубликован
c_4	Внесение изменений в документ
c_5	Утверждение нового документа
c_6	Согласование документа
c_7	Создание локальной копии документа
c_8	Внесение изменений в новую версию документа
c_9	Утверждение новой версии документа
c_{10}	Выпуск документа
c_{11}	Аннулирование документа
c_{12}	Документ не активен

5. После внесения изменений документ переходит в стадию ЖЦ «Утверждение нового документа».

6. При завершении процесса утверждения нового документа происходит его переход в стадию ЖЦ «Согласование документа».

7. В случае успешного согласования документа объявляется окончание его разработки и стадия жизненного цикла изменяется на «Выпуск документа».

8. Если документ не прошел согласование, то необходимо внести в него изменения. Для этого создается локальная копия изменяемого документа (стадия ЖЦ «Создание локальной копии»), затем вносятся изменения в локальную копию документа (стадия ЖЦ «Внесение изменений в новую версию»). После внесения всех необходимых изменений документ переходит в стадию ЖЦ «Утверждение новой версии документа» и передается на согласование.

9. Если документ уже не актуален, то он выводится из проектного цикла и переходит в стадию ЖЦ «Аннулирование документа».

10. Если документ не участвует в этапе проектного цикла, то переходит в стадию ЖЦ «Документ не активен».

Набор операций по изменению стадий ЖЦ документа определен как входной алфавит автомата Мили и представлен в табл. 2.

Представим стадии ЖЦ ЭКД в виде состояний и функций перехода конечного автомата Мили. Описание функций перехода по стадиям ЖЦ ЭКД приведено в табл. 3.

Связав вершины представления конечного автомата дугами в направлении движения от начальной вершины к конечной, согласно табл. 2 и 3, получим модель изменений стадий ЖЦ ЭКД в ЕИП в виде ориентированного графа (рис. 2).

Иницилирующим действием начала работ по ЖЦ конкретного ЭКД является управляющее воздействие как изменение статуса управляемого документа. Признак завершения конкретного

Таблица 2. Операции перехода по стадиям ЖЦ документа

Table 2. Transition operations in the life cycle stages of a document

Стадия ЖЦ документа	Операция перехода в стадию ЖЦ	
	Название	Обозначение
1. Новый	Создать	x_1
2. Опубликован	Опубликовать	x_2
3. Внесение изменений в документ / Внесение изменений в новую версию документа	Изменить	x_3
4. Утверждение нового документа / Утверждение новой версии документа	Утвердить	x_4
5. Согласование документа	Согласовать	x_5
6. Создание локальной копии документа	Копировать	x_6
7. Выпуск документа	Выпустить	x_7
8. Аннулирование документа	Аннулировать	x_8
9. Документ не активен	Вывести	x_9

Таблица 3. Функции перехода по стадиям ЖЦ документа

Table 3. Transition functions in the life cycle stages of a document

Стадия жизненного цикла документа	Состояние	Функция перехода в состояние
1. Новый	c_1	—
2. В разработке	c_2	$x_1 = f(c_1)$ – создать
3. Опубликован	c_3	$x_2 = f(c_2)$ – опубликовать
4. Внесение изменений в документ	c_4	$x_3 = f(c_3)$ – изменить
5. Утверждение нового документа	c_5	$x_4 = f(c_4)$ – утвердить
6. Согласование документа	c_6	$x_5 = f(c_5)$ – согласовать
7. Создание локальной копии	c_7	$x_6 = f(c_6)$ – копировать
8. Изменение новой версии	c_8	$x_3 = f(c_7)$ – выпустить
9. Утверждение новой версии	c_9	$x_4 = f(c_8)$ – утвердить
10. Выпуск документа	c_{10}	$x_7 = f(c_9)$ – выпустить
11. Аннулирование документа	c_{11}	$x_8 = f(c_2)$ – аннулировать
12. Документ не активен	c_{12}	$x_9 = f(c_2)$ – вывести

этапа ЖЦ ЭКД – смена атрибута статуса управляемого документа. Понятие статуса документа трактуется как совокупность атрибутов, однозначно определяющих фазу ЖЦ конкретного документа в рамках моделируемого этапа ПЦ. Условие завершения ЖЦ ЭКД – достижение им стадии c_{10} «Выпущен» или c_{11} «Аннулирован».

Для имитационного моделирования динамики изменения стадий ЖЦ ЭКД используем среду *MATLAB* с подключенным пакетом *Stateflow*. Разработанная имитационная модель включает в себя собственно модель конечного автомата в виде ориентированного графа (рис. 3) и генераторы случайных чисел, определяющие статус документа:

- активен, не нуждается во внесении изменений (0);

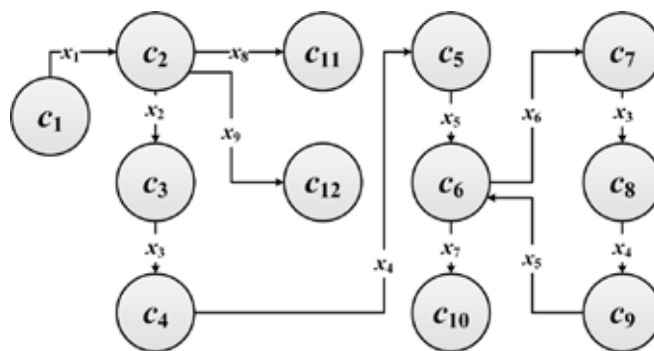


Рис. 2. Модель динамики изменений стадий жизненного цикла ЭКД в ЕИП в виде ориентированного графа

Fig. 2. The model of the change dynamics of the EDD life cycle stages in the UIS in the form of an oriented graph

- активен, нуждается во внесении изменений (1);
- не активен (2);
- аннулирован (3).

Имитационная модель КА представлена тремя отдельными циклами обработки документа (рис. 3):

- 1) обработка без внесения изменений (поз. 1);
- 2) обработка с внесением изменений (поз. 2);
- 3) вывод из обращения (поз. 3).

Номера состояний документа на рис. 3 указаны в кружках и соответствуют номерам состояний, приведенным в табл. 3.

На рис. 4 показана форма отображения динамики выполнения процессов стадий жизненного цикла ЭКД в ходе моделирования в пакете *Stateflow*. Так, в приведенном случае активный ЭКД находится в стадии ЖЦ c_2 «В разработке», при этом не известно, будут в него внесены изменения или нет. Текущая стадия документа в ходе моделирования выделена жирной линией. На рис. 5 активный ЭКД находится в стадии ЖЦ c_9 «Утверждение», при этом в документ уже внесены изменения и в данной стадии выполняется утверждение новой версии документа.

Разработанная имитационная модель позволяет эффективно проводить событийное моделирование организационных и технических процессов, которые сопровождаются изменением статуса электронной документации и, соответственно, нуждаются во взаимном согласовании. В рассмотренном случае моделирование показало потенциальную выполнимость ЖЦ ЭКД, отсутствие бесконечных циклов, «тупиков» и других коллизий, препятствующих протеканию процессов жизненного цикла конструкторского документа.

Формирование интегрированного информационного пространства поддержки проектирования, разработки и проведения испытаний радиоэлектронной аппаратуры

В качестве прикладной области применения предложенных методов и средств организации электронного документооборота и единого информационного пространства рассмотрим интегра-

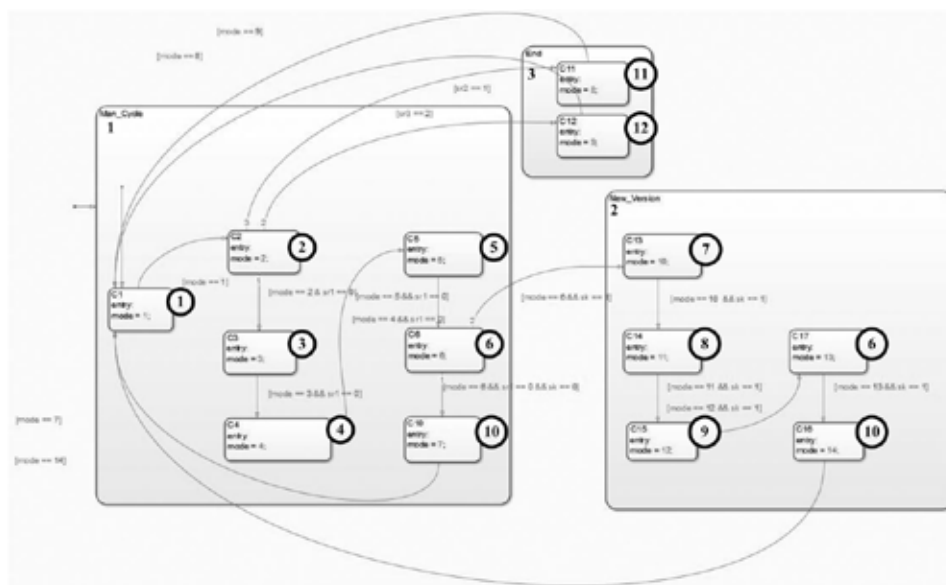


Рис. 3. Имитационная модель динамики изменения стадий ЖЦ ЭКД

Fig. 3. The simulation model of the dynamics of the EDD life cycle stages

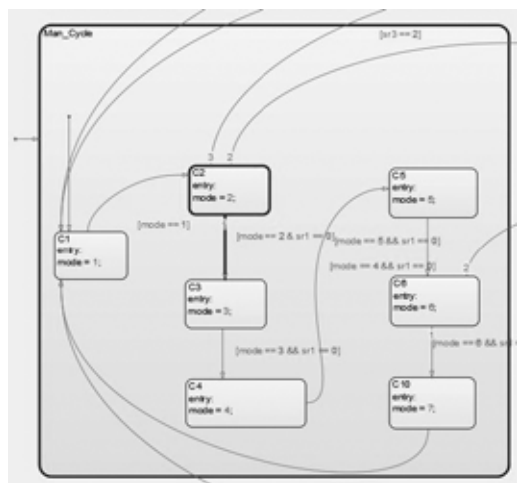


Рис. 4. Отображение стадии ЖЦ ЭКД «В разработке»

Fig. 4. Display of the EDD life cycle stage “Development”

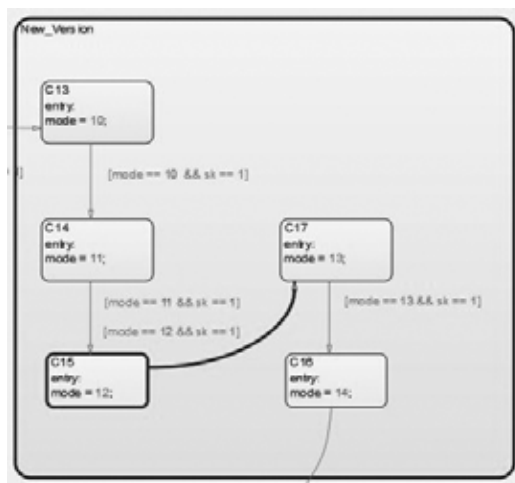


Рис. 5. Отображение стадии ЖЦ ЭКД «Утверждение»

Fig. 5. Display of the EDD life cycle stage “Approval”

цию программных и аппаратных средств, используемых при проектировании цифровых приемников навигационных спутниковых сигналов (ГНСС-приемников) с измерением углов пространственной ориентации подвижных объектов (углов Эйлера). Такого рода устройства предназначены для работы с навигационными сигналами (НС) от двух навигационных космических аппаратов (НКА), реализуются на базе заказных или полужаказных интегральных схем, базовых матричных кристаллов (БМК) или программируемых логических интегральных схем (ПЛИС).

С точки зрения организации процесса проектирования разрабатываемый угломерный ГНСС-приемник можно представить в виде двух составляющих:

- программный имитатор сигналов от НКА (рис. 6, поз. 1);
- структурные единицы ГНСС-приемника (рис. 6, поз. 2).

Разрабатываемый ГНСС-приемник включает в себя составляющие [16, 17]:

- Γ_1 – блок генерации информационной и частотной составляющей НС НКА 1;
- Γ_2 – блок генерации информационной и частотной составляющей НС НКА 2;
- $КА_1$ – блок имитатора генерации и усиления НС на НКА 1;
- $КА_2$ – блок имитатора генерации и усиления НС на НКА 2;
- $СФ_1$ – блок ввода фазового сдвига НС от НКА 1;
- $СФ_2$ – блок ввода фазового сдвига НС от НКА 2;
- $С_1$ – селектор НКА;
- Π_{L1} – приемник НС L1;
- Π_{L2} – приемник НС L2;
- $РН_1$ – блок разрешения фазовой неоднозначности НС с использованием двухчастотного режима;
- Φ_1 – блок вычисления фазового сдвига НС;
- $УО_1$ – блок разрешения системы уравнений определения угловой ориентации объекта (углов Эйлера) по величине фазового сдвига НС.

В целом процесс проектирования цифровой системы по методологии СнК включает в себя следующие этапы: системное проектирование, функциональное проектирование, логическое проектирование, топологическое проектирование, этап верификации и подготовки к производству [18]. Рассмотрим пример интеграции средств проектирования для этапа системного проектирования, сделав допущение, что методы интеграции для остальных этапов ПЦ аналогичны.

Блок-схема этапа системного проектирования угломерного ГНСС-приемника представлена на рис. 7. Разработка СнК на этапе системного проектирования начинается с анализа требований, а также разработки спецификации системы (рис. 7, поз. 1.1). При этом определяются основные эксплуатационно-технические свойства СнК, такие как требуемое быстродействие и допустимая потребляемая мощность. Алгоритм работы СнК на уровне математического описания разрабатывается на основе спецификации системы (рис. 7, поз. 1.2). На данном этапе произ-

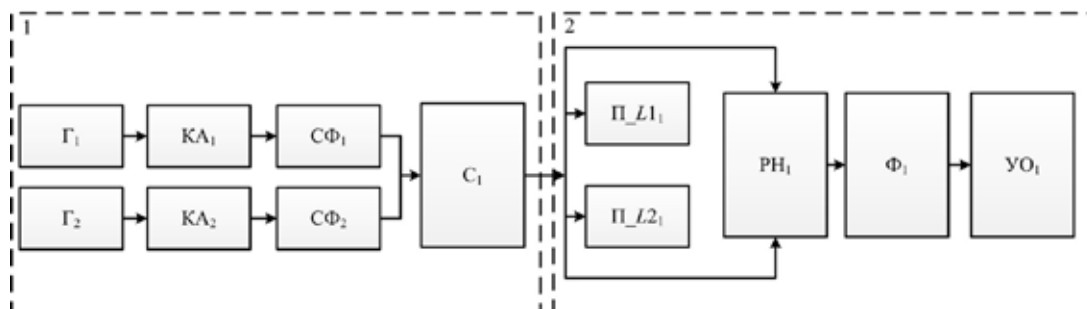


Рис. 6. Структурная схема угломерного ГНСС-приемника

Fig. 6. The block diagram of the angular Global Navigation Satellite System (GNSS) receiver

водится математическое моделирование разработанных алгоритмов функционирования СнК, оцениваются требуемое быстродействие, разрядность и форматы представления внутренних данных.

На основе алгоритма работы разрабатывается поведенческая модель, отражающая общие принципы функционирования СнК (рис. 7, поз. 1.3). Для верификации поведенческой модели разрабатывают поведенческую модель тестового окружения (рис. 7, поз. 1.4). Она включает в себя тестовые последовательности, генераторы входных сигналов и средства отображения выходной информации. На основе разработанной модели впоследствии проектируются и реализуются тестовые последовательности для верификации проекта на нижних уровнях проектирования, а также для тестирования опытных образцов. После этого проводят верификацию поведенческой модели системы (рис. 7, поз. 1.5).

На основе поведенческой модели системы разрабатывается структурная модель системы на уровне функциональных блоков (рис. 7, поз. 1.6), отражающая принципы взаимодействия отдельных блоков в составе СнК и включающая их основные параметры.

Структурная модель системы позволяет выполнить генерацию описания блоков разрабатываемой СнК на уровне регистровых передач с использованием языков описания аппаратуры (*HDL*-описания, рис. 7, поз. 1.7), аналогично генерируется *HDL*-описание тестового окру-



Рис. 7. Блок-схема этапа системного проектирования ГНСС-приемника

Fig. 7. The block diagram of the GNSS receiver system design stage

жения (рис. 7, поз. 1.8). Полученное *HDL*-описание структурной модели системы проходит процедуру программно-аппаратной верификации с использованием аппаратных отладочных средств на базе ПЛИС (рис. 7, поз. 1.9) совместно с программным тестовым окружением, разработанным на этапах 1.4 и 1.8.

Электронные конструкторские документы, создаваемые в ходе этапа системного проектирования, могут быть представлены в различных форматах файлов в зависимости от используемого на конкретном этапе программного обеспечения (табл. 4), в частности:

- *.doc, .docx* – текстовый процессор *MS Word*;
- *.xls, .xlsx* – табличный процессор *MS Excel*;
- *.ns, .msw* – система компьютерной алгебры *Maple*;
- *.eap* – среда проектирования программного обеспечения *Enterprise Architect*;
- *.m, .slx, .html, .xml, .vhd* – среда технических вычислений и моделирования *MATLAB/Simulink*;
- *.vhd, .xise, .ucf, .bit* – системы автоматизированного проектирования *Xilinx ISE Design Suite* и *Vivado*;
- *.vi* – система моделирования и сбора данных *LabVIEW*;
- *.mpf* – среда моделирования и отладки *HDL*-кода *Mentor Graphics ModelSim*.

Таблица 4. Программные средства, которые используются на этапах системного проектирования ГНСС-приемника и порождаемые ими форматы ЭКД

Table 4. Software that is used at the stages of the GNSS receiver system design and EDD formats generated by them

Номер этапа	Используемое программное обеспечение	Форматы документов	Типы документов
1.1	<i>MS Word</i>	<i>.doc, .docx</i>	v_3, v_4
1.2	<i>MATLAB, Maple, Enterprise Architect</i>	<i>.m, .ns, .msw, .eap</i>	v_2, v_5
1.3	<i>MATLAB, Simulink:</i>	<i>.m, .slx</i>	v_1, v_2
1.4	<i>Communications System Toolbox;</i> <i>DSP System Toolbox</i>		
1.5	<i>MATLAB, Simulink:</i> <i>Simulink Control Design;</i> <i>Simulink Design Optimization;</i> <i>Simulink Design Verifier</i>	<i>.html</i>	v_3
1.6	<i>MATLAB, Simulink:</i> <i>HDL Coder;</i> <i>Communications System Toolbox HDL Support</i>	<i>.m, .slx</i>	v_1, v_2
1.7	<i>MATLAB, Simulink:</i>	<i>.m, .slx, .vhd, .html</i>	v_1, v_2, v_3, v_4
1.8	<i>HDL Workflow Advisor;</i> <i>HDL Code Generation</i>		
1.9	<i>LabVIEW, ISE Design Suite, Vivado, ModelSim,</i> <i>MATLAB, Simulink:</i> <i>HDL Verifier;</i> <i>Instrument Control Toolbox;</i> <i>MS Excel</i>	<i>.vhd, .xise, .ucf, .bit,</i> <i>.vi, .mpf, .html,</i> <i>.xls, .xlsx</i>	v_2, v_3, v_4, v_4

Также у каждого ЭКД имеется статус происхождения: «извне», «создается», «создается/передается». Каждый из создаваемых ЭКД относится к одной из категорий документов v_k .

Проектирование на системном этапе осуществляется с помощью программно управляемого автоматизированного стенда разработки и проведения испытаний, включающего в себя соответствующие аппаратные и программные средства. Информационное взаимодействие программно-аппаратной структуры в составе такого стенда приведено на рис. 8.

Ядром автоматизированного программно-управляемого стенда является персональная ЭВМ с развернутыми на ней программными средствами проектирования и разработки (рис. 8, поз. 6), включающими в себя прикладное программное обеспечение (рис. 8, поз. 7) и программный модуль преобразователя в единый формат ЭКД (рис. 8, поз. 8), через который осуществляется взаимодействие с PDM-системой (рис. 8, поз. 11). Аппаратные средства разработки включают в себя измерительное оборудование (генераторы сигналов, осциллографы, анализаторы спектра, рис. 8, поз. 1) и отладочные средства (отладочные платы и комплекты разработчика, рис. 8, поз. 9).

Взаимодействие между программными средствами и измерительным оборудованием осуществляется посредством программной архитектуры виртуальных устройств (VISA, рис. 8, поз. 5), использующей стандартизированный интерфейс ввода-вывода тестирования и проведения измерений для управления приборами, поддерживающими интерфейсы IEEE-488 (GPIB, рис. 8, поз. 2), USB 2.0 (рис. 8, поз. 3) и LAN (рис. 8, поз. 4) измерительных устройств. Взаимодействие между отладочными средствами и программным обеспечением осуществляется посредством шины ввода-вывода PCI (рис. 8, поз. 10).

Со стороны программных средств за обеспечение взаимодействия отвечают:

- с измерительным оборудованием – встроенные модули среды *LabVIEW* и модуль расширения *Instrument Control Toolbox* среды *MATLAB/Simulink*;

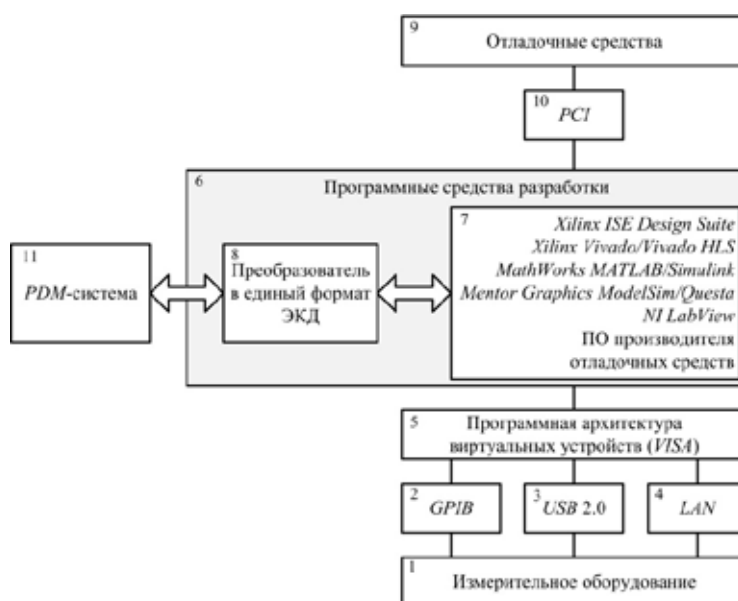


Рис. 8. Схема информационного взаимодействия аппаратных и программных средств разработки

Fig. 8. The scheme of information interaction between hardware and software development tools

- с отладочными средствами – программное обеспечение производителя отладочных средств *DIMECheck* и *FUSE Probe*.

С точки зрения аппаратной реализации автоматизированный стенд включает в себя следующие компоненты:

- высокопроизводительная персональная ЭВМ;
- отладочное средство *Nallatech Xtreme DSP Development Kit-IV*, включающее в себя ПЛИС *Xilinx Virtex-IV XC4VSX35-10FF668* и *Virtex-II XC2V80-4CS144*;
- генератор сигналов произвольной формы *GW Instek SFG-2108*;
- цифровой осциллограф *LeCroy WaveRunner 64Xi*,
- анализатор спектра сигнала *Agilent FieldFox Analyzer N9914A* (рис. 9).

Программная реализация моделей структуры и динамики ЖЦ ЭКД выполнена на базе системы управления инженерными данными «Лощман: *PLM*». В качестве унифицированного формата ЭКД разрабатываемого изделия использованы открытые стандарты обмена данными на базе расширенного языка разметки *XML*. При невозможности экспорта информации в едином формате берут конвертор, преобразующий текстовую информацию в таблицу формата *.xls*. Структура разрабатываемого угломерного ГНСС-приемника дана в ЕИП в виде схематического описания *IP-ХАСТ* (стандарт *IEEE 1685*), представляющего собой расширение языка *XML* для документирования метаданных блоков, применяется в проектировании, разработке и верификации электронных систем [19].

Преобразователь в единый формат ЭКД выполнен на языке *C#* с использованием скриптового языка высокого уровня *Tcl*. Взаимодействие преобразователя со средой *LabVIEW* осуществляется с помощью расширения *Model Interface Toolkit*, со средой *MATLAB* – с использованием модуля *tclmatlab* свободно распространяемой среды разработки *Tloona*, ориентированной на язык *Tcl*. Взаимодействие преобразователя с прочими программными продуктами осуществляется с помощью встроенных интерпретаторов языка *Tcl*.

Разработанный автоматизированный стенд обеспечивает выполнение процессов проектирования, разработки и проведения испытаний ГНСС-приемника на протяжении всего проек-

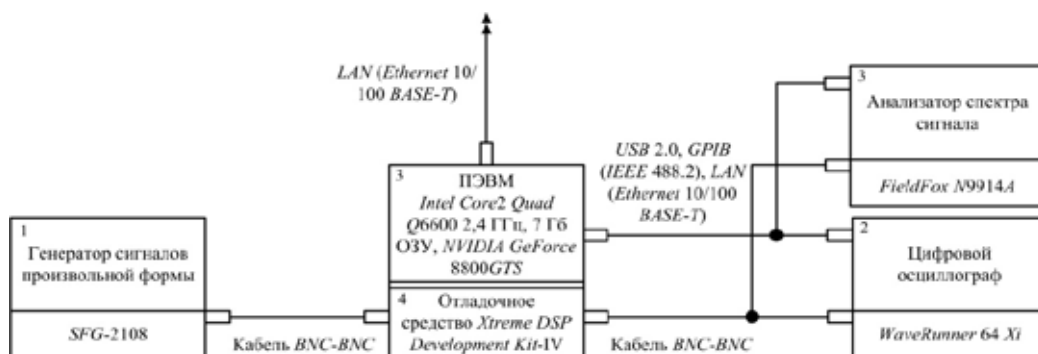


Рис. 9. Структура аппаратной реализации автоматизированного стенда проектирования, разработки и проведения исследовательских испытаний

Fig. 9. The hardware implementation structure of the automated stand for design, development and conduct of research tests

ного цикла. При этом реализовано автоматизированное протоколирование результатов испытаний и формирование требуемого комплекта конструкторской документации в электронном виде. Комплект ЭКД, созданный с применением предложенных методов и алгоритмов, обладает требуемой полнотой и непротиворечивостью.

В качестве примера на рис. 10–12 приведены диаграмма спектра, автоматизированные протоколы исследовательских испытаний угломерного ГНСС-приемника, полученные с использованием разработанного стенда.

Таким образом, автоматизированный стенд для проектирования, разработки и исследования ГНСС-приемников позволяет организовать выполнение процессов проектного цикла радиоэлектронной аппаратуры заданного назначения в условиях автоматизации обработки информации, накопления результатов различных итераций процессов проектирования и протоколирования результатов исследовательских испытаний. Накопление информации и ее передача

ИНФОРМАЦИЯ (3 кадр)		
Географические координаты		
Широта	56°0.3995'N	
Долгота	92°50.5944'E	
Высота	144.33м	
Геом.Фактор	1.30	
Геоцентрические координаты		
X	-177298.3 м	
Y	3563907.9 м	
Z	5264975.7 м	
X проекции Г-К	6209350.6 м	
Y проекции Г-К	16490233.0 м	
Угловые координаты		
Азимут	210.127	
Крен	-0.057	
Угол места	-0.337	
UFlag	5	
Время и частота		
МПК-UTC	6.617618 мс	
ГЛОНАСС-GPS	288.389276 нс	
Расстройка ОГ	1.039E-010	
Отстройка ВхМВ	0 нс	
Количество НКА в расчёте: 9 ГЛОНАСС и 9 GPS		
Режим навигации: Готовность координат: 0%		
29 Апрель 2013 11:54:28		

Рис. 10. Настройки автоматизированного стенда для проведения испытаний ГНСС-приемника

Fig. 10. Automated stand settings for GNSS receiver testing

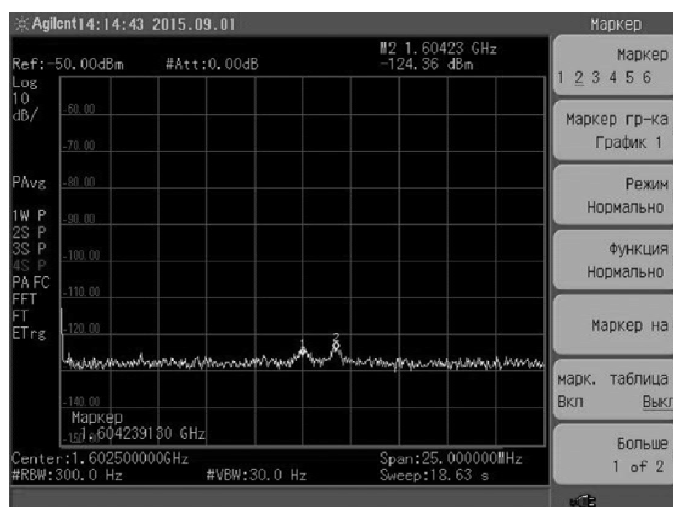


Рис. 11. Диаграмма спектра навигационного сигнала ГНСС-приемника

Fig. 11. The Spectrum Chart of the GNSS Receiver Navigation Signal

	Число спутников	% наблюдения
12	11	0
13	12	0
14	13	0
15	14	0
16	15	0
17	16	0.0261165
18	17	0.504919
19	18	4.71838
20	19	15.5045
21	20	32.2451
22	21	32.6891
23	22	14.1725
24	23	0.139288
25	>=3	100

	Число спутников	% наблюдения
1	0	6.92957
2	1	51.2144
3	2	18.8996
4	3	19.6657
5	4	3.29068
6	5	0

Рис. 12. Протоколы измерений видимости для группировки ГЛОНАСС

Fig. 12. Protocols of visibility measurement for the GLONASS satellite group

обеспечиваются в стандартных форматах – *XML* или *xls*. Хранилище (репозиторий) информации реализовано на основе комплекса обработки инженерной информации «Лоцман: *PLM*».

Заключение

Разработка современных электронных устройств на основе интегральных схем осуществляется в сложной информационной инфраструктуре, включающей средства автоматизированного проектирования, библиотеки логических элементов и функциональных блоков, технологические библиотеки, средства для расчета коррекций на фотошаблонах, средства и методы подготовки изделий к производственному контролю. Микроэлектронные дизайн-центры и фабрики, реализующие единую согласованную информационную инфраструктуру, получили название «разумных» производств. «Разумное» производство включает в себя не только технологический участок, но и центр проектирования и прототипирования, службу поддержки и согласования требований заказчиков. При этом допускается использование уникальных согласованных модификаций технологического процесса и маршрутов проектирования для каждого проекта. Такое производство существенно дороже, требует организации «сквозного» процесса проектирования и производства, отличается наличием системы электронного конструкторского документооборота, единого хранилища непротиворечивой информации. Это способствует созданию единого информационного пространства, обеспечивает эффективность процесса проектирования и последующий рентабельный выпуск малой серией уникальных изделий с высокой добавленной стоимостью.

В работе предложена гибридная модель электронного конструкторского документа в форме сочетания теоретико-множественной, графовой и автоматной моделей с использованием аппарата предметно-ориентированных онтологий, включающая в себя модели структуры и динамики жизненного цикла электронного конструкторского документа. Предложен подход к ин-

теграции аппаратных и программных средств разработки с использованием преобразователя-конвертора.

В целом создание единого информационного пространства для поддержки проектирования, разработки и проведения испытаний радиоэлектронной аппаратуры стоит начинать с определения требований к единому формату конструкторской документации, формализации стадий и этапов процессов проектирования. Такой подход позволяет:

- обеспечить пониженный уровень информационной неопределенности, неизбежный в процессе проектирования любого сложного устройства или системы;
- снизить влияние человеческого фактора при проектировании и проведении исследований за счет автоматизированного заполнения и обработки метаданных в конструкторской документации и протоколах испытаний;
- объединить отдельные программно-аппаратные средства поливендорной среды в единое пространство и, как следствие, повысить эффективность использования имеющихся платформ, обеспечить снижение затрат на их доработку или разработку дополнительных.

Стоит отметить, что в представленном варианте используется *PLM*-система, ориентированная на металлоемкие отрасли машиностроения с преобладанием крупносерийного производства. Для более эффективного ее использования при информатизации проектной деятельности необходима модификация исходной системы под требования приборостроительной отрасли. Также возможна реализация отдельных функций *PLM/PDM*-системы на базе систем контроля версий, таких как *Rational ClearCase*, *Perforce* и *PVCS*, получивших широкое распространение при организации разработки программного обеспечения.

Работа выполнена при финансовой поддержке Министерства образования и науки РФ (договор № 02.G25.31.0202 от 27 апреля 2016 г.).

Список литературы

- [1] Wei H., Hong G., Hongna G., Kai Z., Jun L., Xiaoming L. A Novel Design of Software System on Chip for Embedded System, *Journal of Signal Processing Systems*, 2017, 86 (2), 135–147.
- [2] Feero B.S. Networks-on-chip in a three-dimensional environment: a performance evaluation, *IEEE Transactions on Computers*, 2009, 58 (1), 32–45.
- [3] Хюбнер М., Бекер Ю. и др. *Многопроцессорные системы на одном кристалле. Разработка аппаратных средств и интеграция инструментов*. М.: Техносфера, 2012. 304 с. [Hübner M., Becker J. *Multiprocessor System-on-Chip. Development of hardware and integration of tools*. Moscow, Tekhnosfera, 2012, 304 p. (in Russian)]
- [4] Борискин В.С., Гулякович Г.Н., Северцев В.Н. Организация мелкосерийного производства микросхем. *Инженерный вестник Дона*, 2012, 20 (2), 310–314. [Boriskin V.S., Gulyakovich G.N., Severtsev V.N. Organization of small batch production of chips, *Engineering journal of Don*, 2012, 20 (2), 310–314. (in Russian)]
- [5] Загидуллин Р.Р. *Управление машиностроительным производством с помощью систем MES, APS, ERP: Монография*. Старый Оскол: ТНТ, 2011. 372 с. [Zagidullin R.R. *Management of engineering production with the help of MES, APS, ERP: Monograph*. Stary Oskol: TNT, 2011, 372 p. (in Russian)]

- [6] Hu H., Wang L., Luh P. Intelligent manufacturing: New advances and challenges, *Journal of Intelligent Manufacturing*, 2015, 26 (5), 841–843.
- [7] Duy Dao S., Abhary K., Marian R. An innovative model for resource scheduling in VCI systems, *Operational Research*, 2016, 1–22.
- [8] Iizuka M., Hamada N., Saito H., Yamaguchi R., Yoshinaga M. A tool set for the design of asynchronous circuits with bundled-data implementation, *IEEE 29th International Conference on Computer Design (ICCD)*, 2011, 78–83.
- [9] Musa A., Gunasekaran A., Yusuf Y. Supply chain product visibility: Methods, systems and impacts, *Expert Systems with Applications*, 2014, 41, 176–194.
- [10] Mao Xi, Li Qi, Zhang Ziming Subject-oriented spatial information search engine based on ontology, *Proceedings of Information Technology and Environmental System Science (ITESS)*, 2008, 3, 843–848.
- [11] Lavrishcheva E. Ontological Approach to the Formal Specification of the Standard Life Cycle, *Proceedings of the Science and Information Conference (SAI)*, 2015, 965–972.
- [12] Соловьев В.Д., Добров Б.В., Иванов В.В., Лукашевич Н.В. *Онтологии и тезаурусы: модели, инструменты, приложения*. М.: Бином. Лаборатория знаний, 2009. 173 с. [Solovov V.D., Dobrov B.V., Ivanov V.V., Lukashevich N.V. *Ontologies and thesauruses: models, tools, application*. Moscow: Binom. Laboratoriya znaniy, 2009, 173 p. (in Russian)]
- [13] Липаев В.В. *Человеческие факторы в программной инженерии: рекомендации и требования к профессиональной квалификации специалистов*. М.: СИНТЕГ, 2009. 328 с. [Lipaev V.V. *Human factors in software engineering: recommendations and requirements for the professional qualification of specialists*. Moscow: SINTEG, 2009, 328 p. (in Russian)]
- [14] Evans M., Maglaras L.A., He Y., Janicke H. Human behaviour as an aspect of cybersecurity assurance, *Security and Communication Networks*, 2016, 9 (17), 4667–4679.
- [15] Gopalakrishna A.K., Ozcebe T., Lekkien J.J., Liotta A. Relevance in cyber-physical systems with humans in the loop, *Concurrency and Computation-Practice and Experience*, 2017, 29 (3), 1–18.
- [16] Перов А.И., Харисов В.Н. и др. *ГЛОНАСС. Принципы построения и функционирования*. М.: Радиотехника, 2010. 800 с. [Perov A.I., Harisov V.N. et al. *GLONASS. Principles of construction and operation*. Moscow: Radiotekhnika, 2010, 800 p. (in Russian)]
- [17] Перов А.И. *Основы построения спутниковых радионавигационных систем*. М.: Радиотехника, 2012. 240 с. [Perov A.I. *Basics of building satellite radio navigation systems*. Moscow: Radiotekhnika, 2012, 240 p. (in Russian)]
- [18] Наваби З. *Проектирование встраиваемых систем на ПЛИС*. М.: ДМК Пресс, 2016. 464 с. [Navabi Z. *Embedded Core Design with FPGAs*. Moscow, DMK Press, 2016, 464 p. (in Russian)]
- [19] Matilainen L., Lehtonen L., Maatta J.M., Salminen E., Hamalainen T.D. System-on-Chip deployment with MCAPI abstraction and IP-XACT metadata, *Samos XII, International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation, Samos: IEEE*, 2012, 209–216.